rti-s
HLA Prototype Software for STOW

AMG 17

February 13, 1997

Jim Calvin, jcalvin@ll.mit.edu

Outline

- Sponsors
- STOW System Requirements
- Key Issues
- Approach
- Design
- Next Steps

rti-s Sponsors

- Defense Modeling and Simulation Office (DMSO)
  - Part of larger DoD 1.0 RTI development effort
  - Initial implementation of Data Distribution Management Services
  - Support initial use of the HLA in a a large scale, real-time system
- DARPA
  - Synthetic Theater of War (STOW) program
    - STOW is an Advanced Concept Technology Demonstration (ACTD)
    - Customer is Atlantic Command (ACOM)
    - An HLA based simulation system

Some STOW Requirements

(RTI perspective)

- ~25 Federates
- 350 – 450 host computers (each a federate instance)
- 10 – 20 sites (LANs connected via a WAN)
- 10,000 objects (scaleable to 100,000)
- Low latency, High throughput, Low bandwidth overhead
- Conserves resources (multicast groups)
- Limit scope of changes within pre-existing applications
- Early availability of RTI (15 Oct 96)

STOW Requirements for RTI Services

Federation Management
- Pause/Resume, Save/Restore, coordination only

Declaration Management

Object Management

Time Management
- Not required

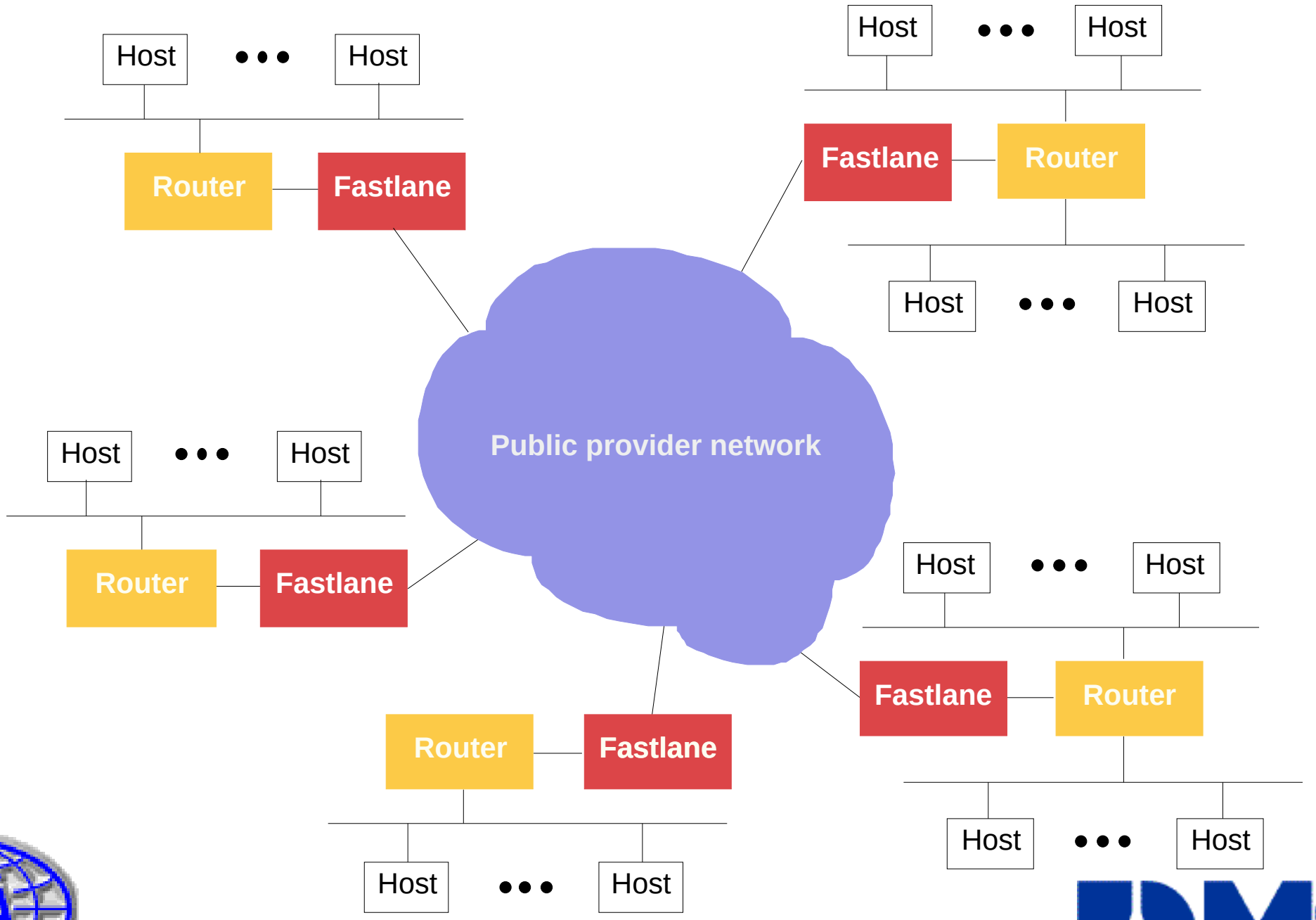
Ownership Management
- Not required, later discovered it would have been usef

Data Distribution Management
- Too many objects, class filtering insufficient
- Implemented and heavily utilized

STOW Requirements for RTI Services

- System monitoring/reporting
  - MOM approach used
- Additional features required by an “operational” system
  - Logging support
  - Support for exercise set up

STOW Network Architecture (simplified)



Key Issues

- Recognize that the STOW requirements present a system engineering problem
- The problem is *not solved* by a set of isolated solutions to individual issues; a systemic approach is required
- Efficiency is crucial
- Resiliency to problems and errors paramount
- Minimum operator intervention for setup, recovery, etc.

Key Issues, continued

- Simulation code is legacy (> 500,000 SLOC)
  – limit the scope of changes
- Limit arrival rate of packets to individual host computers
  – estimated aggregate rate ~12,000pps
- Must run on multiple platforms (Sun Solaris, IRIX 5.3, 6.2, X86 Linux, Solaris X86)
- Limited time and resources to address all requirements and issues
- System to be operational in summer, 1997
- ACTD in November 1997

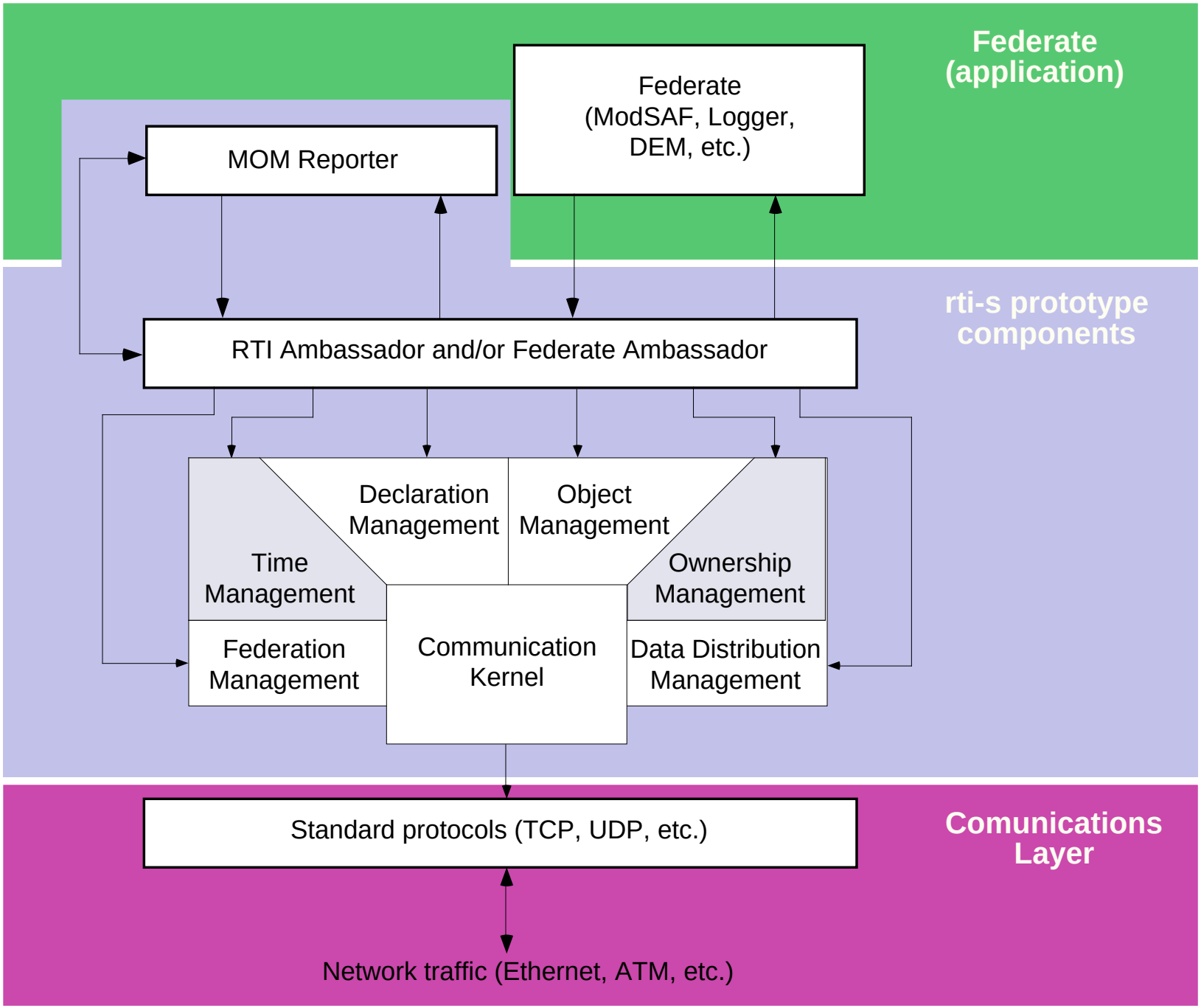
Approach

- Reduce risk for the ACTD
  - Utilize uncoordinated distributed algorithms
    - Upside: robust to network problems, federate and/or processor faults
    - Downside: harder to build than centralized versions, may sacrifice optimality
  - Maximize
    - Lessons and code from 0.x RTI series, RITN, etc.
    - Routing space exploitation
    - RTI throughput
  - Minimize
    - Processor requirements to run RTI
    - Arrival of packets at the host interface

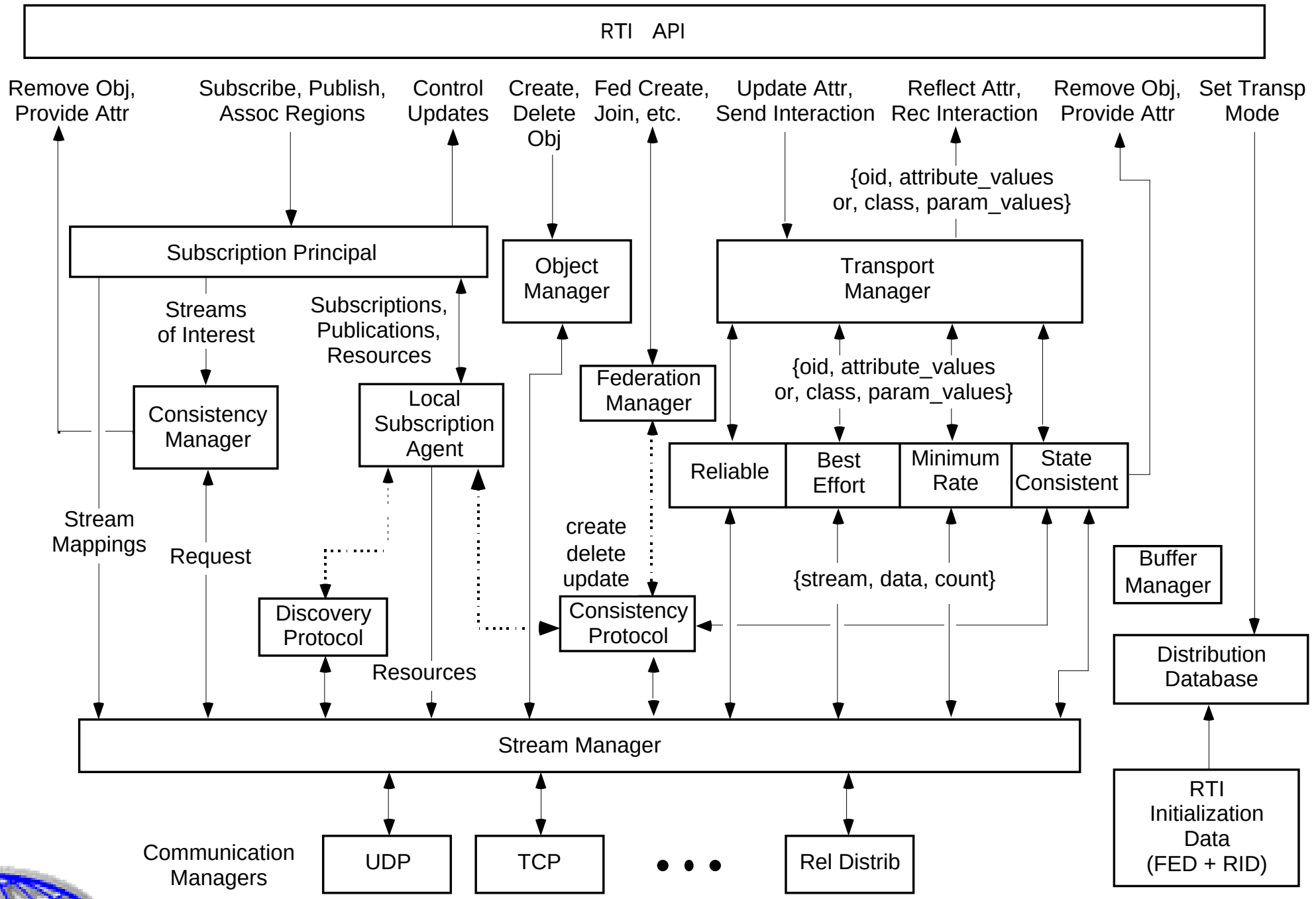
Use of Routing Spaces

- STOW currently defines 16 routing spaces
- Range from one to three dimensions
  - Most are geographic based
  - One includes “ground, low-air, hi-air, ocean”
  - One based on communication “band”
  - Some are essentially class based
- Definition is an essential part of FOM for STOW

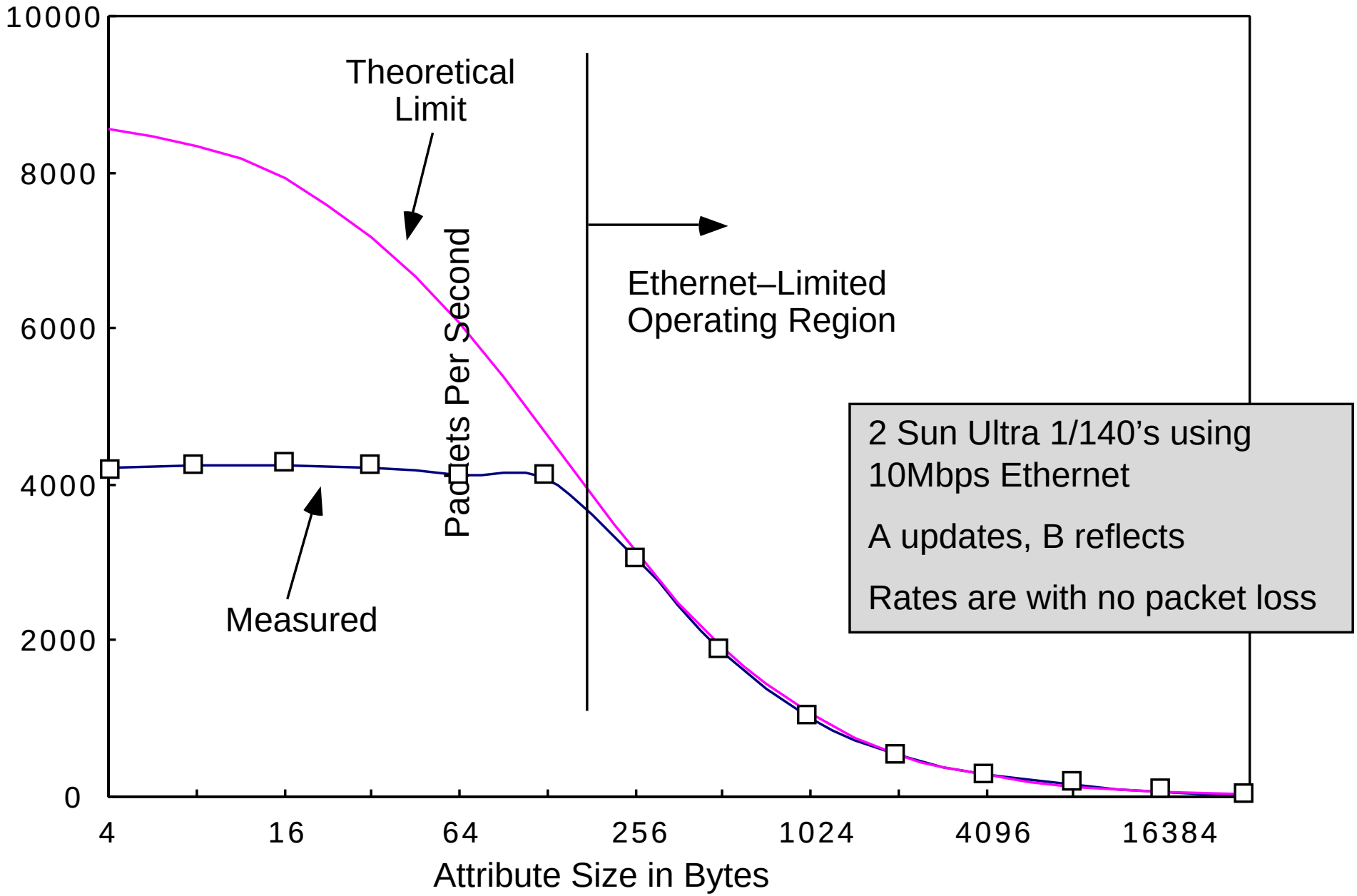
STOW Application Block Diagram



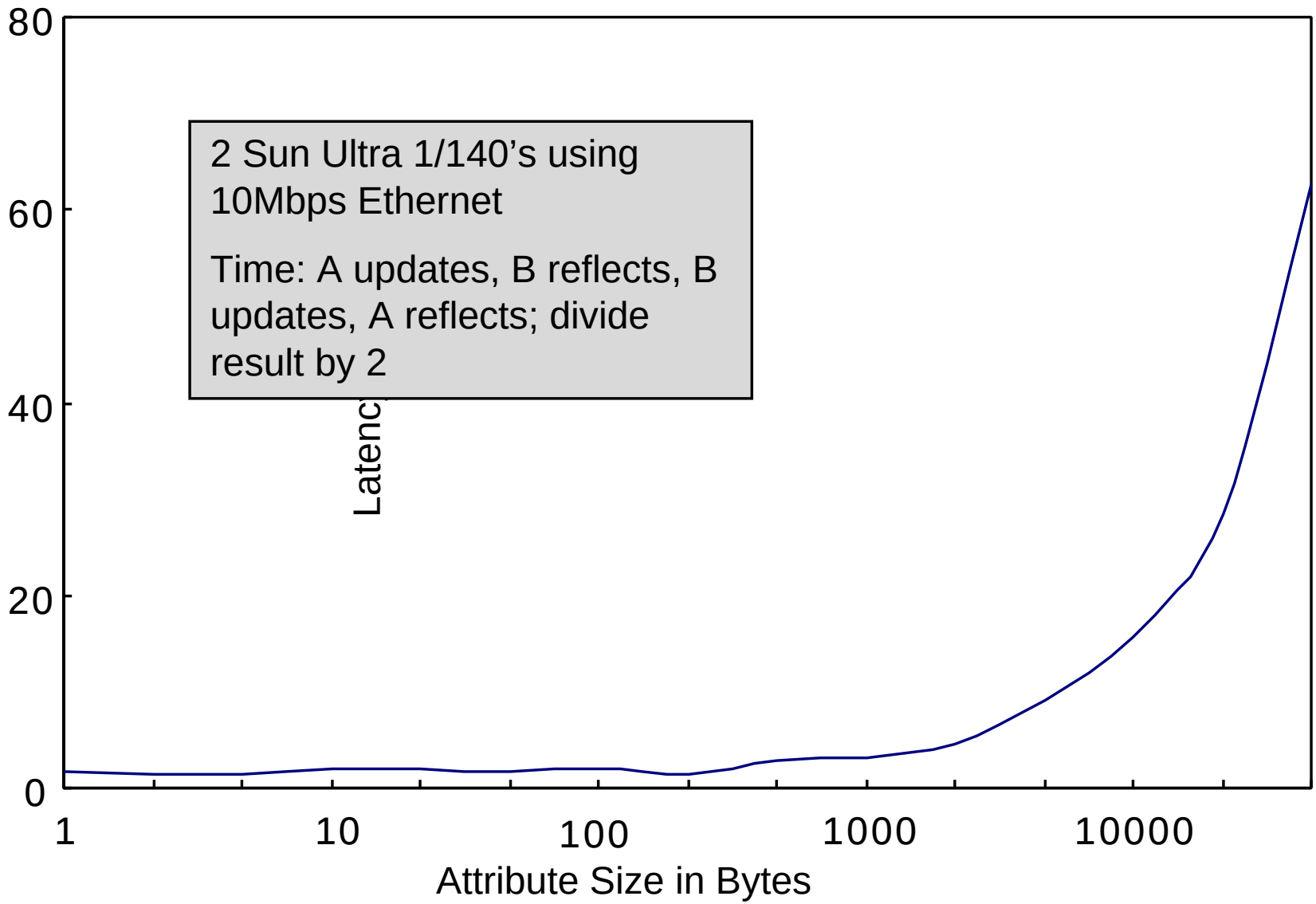
rti-s Block Diagram



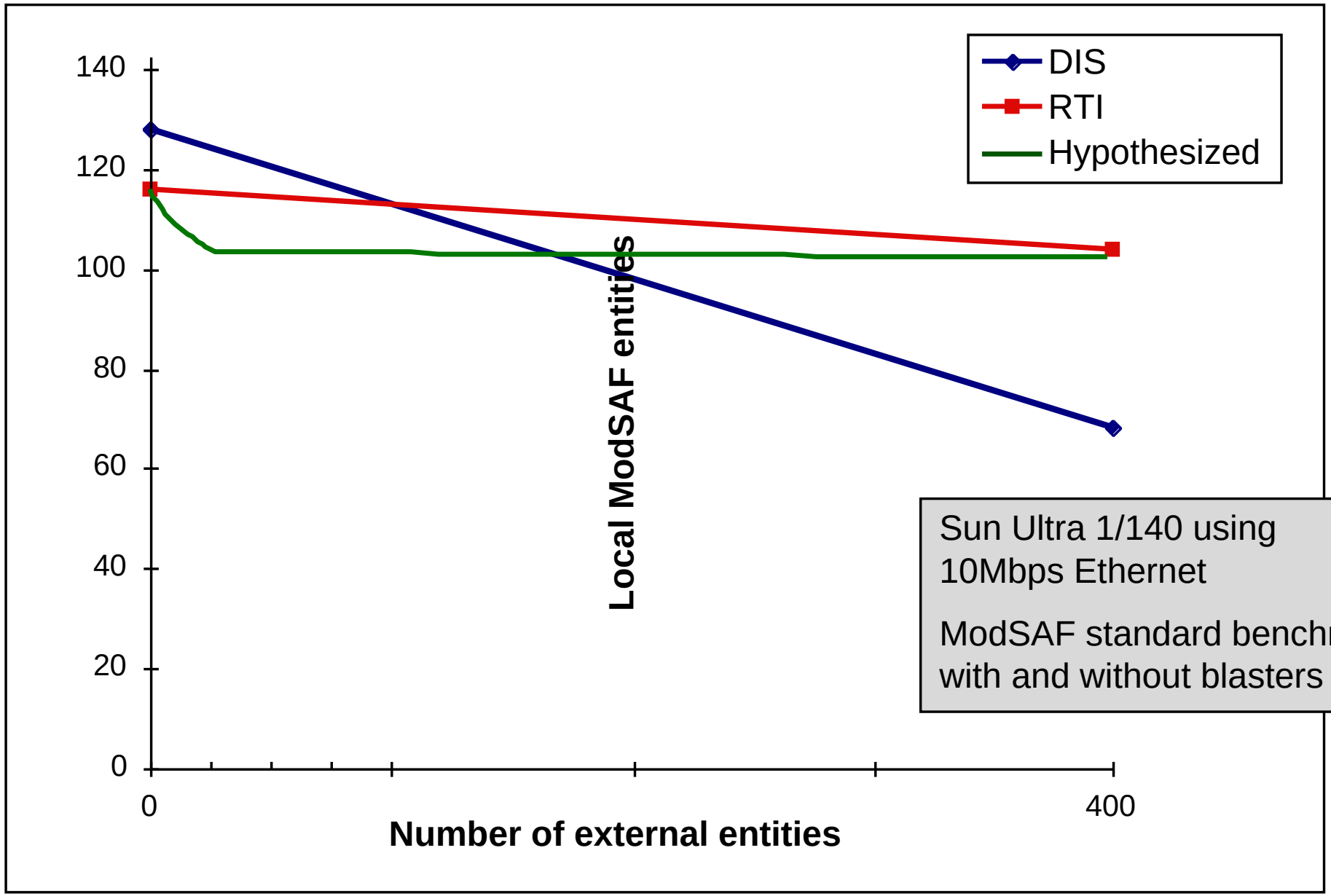
Throughput vs. packet size



Latency vs. Packet Size



Reducing packet arrivals
→ more processor time for simulation



Next steps

- Deliver rti-s/C to STOW, 24 March 97
- Merge rti-s with RTI 1.0 to create RTI 1.1
- Document STOW rti-s experiments
  - Present lessons learned
  - Decide which are generally useful
  - Present those to the AMG for consideration